IN THE CLAIMS

1.   (Cancelled)

2.   (Currently Amended) A method comprising:

selecting one or more software stage objects from an object-oriented programming model; wherein the software stage objects encapsulate and represent functionality performed by underlying hardware to process a packet, the software stage objects including:

one or more link stage objects to define a physical interface and packet framing,

one or more classifier stage objects to direct filtering and matching algorithms on packets,

one or more packet flow stage objects to direct packet flow policy,

a scatter stage object to direct packet routing to scatterer object outputs,

a gather stage object to direct packet collecting and routing scattered packets,

an editor stage object to direct packet modification, and

a monitor stage object to direct gathering of statistical information regarding packets and packet flows;

programming the one or more selected software stage objects to perform a desired packet processing functionality; and

connecting the one or more programmed software stage objects to form a directed graph of packet flow to complete definition of the desired packet processing functionality, such that underlying hardware is directed to process packets in accordance with the desired packet processing functionality.

3.   (Currently Amended) The method of claim 2, wherein the connecting the one or more programmed software objects further comprises:

selecting a firstan input link stage object as an input port of the directed graph to direct a physical interface and packet framing;

selecting a second classifier stage object coupled to the first stage object to direct filtering and matching algorithms on packets;

selecting a third scatter stage object coupled to the second stage object to direct routing of packets to one or more third stage object outputs;

selecting a fourth monitor stage object coupled to an output from the one or more third scatter stage object outputs to direct gathering of statistical information regarding packets and packet flows;

selecting an fifth editor stage object coupled to an output from the one or more third scatter object outputs to direct packet modification;

selecting a ~~sixth~~ gather stage object coupled to an output from the one or more third object outputs to direct packet routing to a sixth stage object output; and

selecting a ~~final~~ an output link stage object as an output port of the directed graph, the ~~final~~ output link stage object coupled to the ~~sixth~~ gather stage object output.

4. (Currently Amended) The method of claim 2~~2~~, wherein the directing underlying hardware further comprises:

relaying requests from the programmed software stage objects contained in the direct~~ed~~ graph to underlying packet forwarding hardware in accordance with the desired packet processing functionality; and

processing packets by the packet forwarding hardware in response to the software stage object requests, such that the directed graph of programmed stage software objects control packet data flow through the packet forwarding hardware.

5. (Cancelled)

6. (Currently Amended) The method of claim ~~5~~2, wherein the one or more software stage objects each include one or more inputs and one or more outputs enabling formation of compositions of objects sharing a common interface to direct packet processing as a group of software object objects and to form directed graphs of software object objects to direct packet data flow through packet forwarding hardware.

7. (Currently Amended) A machine-readable medium having stored thereon data representing sequences of instructions, the sequences of instructions which, when executed by a processor, cause the processor to performing a method comprising:

directing underlying hardware loaded with a directed graph of programmed software stage objects to process packets in a manner specified using an object-oriented model and compiled to produce the directed graph of programmed software stage objects, wherein the software stage objects encapsulate and represent functionality performed by underlying hardware to process a packet, the software stage objects including:

one or more link stage objects to define a physical interface and packet framing,
one or more classifier stage objects to direct filtering and matching algorithms on packets,
one or more packet flow stage objects to direct packet flow policy,
a scatter stage object to direct packet routing to scatterer object outputs,
a gather stage object to direct packet collecting and routing scattered packets,

an editor stage object to direct packet modification; and

a monitor stage object to direct gathering of statistical information regarding packets and packet flows.

8.   (Cancelled)

9.   (Cancelled)

10.   (Currently Amended) The machine-readable medium of claim 8̶7, wherein the directing underlying hardware further comprises:

relaying requests from the software stage objects contained in the direct̶ed graph to underlying packet forwarding hardware in accordance with a desired packet processing functionality; and

performing packet processing by the packet forwarding hardware in response to the software stage object requests, such that the directed graph of programmed software stage objects control packet data flow through the packet forwarding hardware.

11.   (Cancelled)

12.   (Cancelled)

13.   (Currently Amended) An apparatus, comprising:
a processor; and
a memory coupled to the processor, the memory to load a directed graph of programmed software stage objects to direct the processor to process packets in a manner specified using an object-oriented model and compiled to produce the directed graph of programmed software stage objects ,wherein the software stage objects encapsulate and represent functionality performed by underlying hardware to process a packet, the software stage objects including:

one or more link stage objects to define a physical interface and packet framing,

one or more classifier stage objects to direct filtering and matching algorithms on packets,

one or more packet flow stage objects to direct packet flow policy,

a scatter stage object to direct packet routing to scatterer object outputs,

a gather stage object to direct packet collecting and routing scattered packets,

an editor stage object to direct packet modification, and

a monitor stage object to direct gathering of statistical information regarding packets and packet flows.

14. (Currently Amended) The apparatus of claim 13, wherein the memory is further to relay requests from the software stage objects contained in the directed graph to the processor, such that processor processes packets in response to the software stage object requests to control packet data flow through the processor.

15. (Previously Presented) The apparatus of claim 13, wherein the processor comprises a network processor.

16. (Previously Presented) The apparatus of claim 13, wherein the processor comprises an application specific integrated circuit.

17. (Currently Amended) A system comprising:
a wide area network;
a local area network; and
a processor coupled between the wide area network and the local area network on to form a network, the processor having
a memory coupled to the processor, the memory to load a directed graph of programmed software stage objects to direct the processor to process packets in a manner specified using an object-oriented model and compiled to produced the directed graph of programmed software stage objects wherein the software stage objects encapsulate and represent functionality performed by underlying hardware to process a packet, the software stage objects including:
one or more link stage objects to define a physical interface and packet framing,
one or more classifier stage objects to direct filtering and matching algorithms on packets,
one or more packet flow stage objects to direct packet flow policy,
a scatter stage object to direct packet routing to scatterer object outputs,
a gather stage object to direct packet collecting and routing scattered packets,
an editor stage object to direct packet modification, and
a monitor stage object to direct gathering of statistical information regarding packets and packet flows.

18. (Currently Amended) The system of claim 17, wherein the memory is further configured to relay requests from the software stage objects contained in the directed graph to the

processor, such that processor process packets in response to the software <u>stage</u> object requests for controlling packet data flow through the processor.

19. (Previously Presented) The system of claim 17, wherein the processor comprises a network processor.

20. (Previously Presented) The system of claim 17, wherein the processor comprises an application specific integrated circuit.

21. (Currently Amended) The system of claim 17, wherein each software <u>stage</u> object within the directed graph performs a data-path packet processing task functionality, such that the directed graph of programmed software <u>stage</u> objects performs a plurality of data-path packet ~~procession~~ <u>processing</u> tasks within a single device.

22. (Currently Amended) The method of claim 2, further comprising:
loading underlying hardware with the directed graph of programmed software <u>stage</u> objects to process packets in accordance with the desired packet processing functionality.

23. (Currently Amended) The method of claim 4:
wherein each software <u>stage</u> object within the directed graph performs a data-path packet processing task functionality, such that the directed graph of programmed software objects performs a plurality of data-path packet processing tasks within a single device.

24. (Currently Amended) The machine readable medium of claim ~~10~~<u>7</u>,
wherein each software <u>stage</u> object within the directed graph performs a data-path packet processing task functionality, such that the directed graph of programmed software objects performs a plurality of data-path packet processing tasks within a single device.

25. (Currently Amended) A method comprising:
directing underlying hardware loaded with a directed graph of programmed software <u>stage</u> objects to process packets in a manner specified using an object-oriented model and compiled to produce the directed graph of programmed software <u>stage</u> objects <u>wherein the software stage objects encapsulate and represent functionality performed by underlying hardware to process a packet, the software stage objects including:</u>
<u>one or more link stage objects to define a physical interface and packet framing,</u>

one or more classifier stage objects to direct filtering and matching algorithms on packets,

        one or more packet flow stage objects to direct packet flow policy,

        a scatter stage object to direct packet routing to scatterer object outputs,

        a gather stage object to direct packet collecting and routing scattered packets,

        an editor stage object to direct packet modification, and

        a monitor stage object to direct gathering of statistical information regarding packets and packet flows.

26.   (Currently Amended) The method of claim 25, wherein the directing underlying hardware further comprises:

    relaying requests from the programmed software stage objects contained in the directed graph to underlying packet forwarding hardware in accordance with the desired packet processing functionality; and

    processing packets by the packet forwarding hardware in response to the software stage object requests, such that the directed graph of programmed software objects control packet data flow through the packet forwarding hardware.

27.   (Currently Amended) The method of claim 26, wherein each software stage object within the directed graph performs a data-path packet processing task functionality, such that the directed graph of programmed software objects performs a plurality of data-path packet processing tasks within a single device.